VI International Conference on Computational Methods for Coupled Problems in Science and Engineering
COUPLED PROBLEMS 2015
B. Schrefler, E. Oñate and M. Papadrakakis(Eds)

IMPLEMENTATION OF AN EFFICIENT COUPLED FEM-SBFEM APPROACH FOR SOIL-STRUCTURE-INTERACTION ANALYSIS

Marco Schauer*, Sabine C. Langer†

*Technische Universität Braunschweig
Institut für Statik
Beethovenstrae 51, 38106 Braunschweig, Germany
e-mail: m.schauer@tu-braunschweig.de, web page: https://www.tu-braunschweig.de/statik

†Technische Universität Braunschweig
Institut für Konstruktionstechnik
Langer Kamp 8, 38106 Braunschweig, Germany
e-mail: s.langer@tu-braunschweig.de, web page: https://www.tu-braunschweig.de/ik

Key words: scaled boundary finite element method, model reduction, soil-structure-interaktion

Abstract. Buildings are grounded in the surrounding soil, so that soil and structure interact with each other. Consequently in the soil induced vibrations are transmitted to the structures. Neighbouring buildings and structures interact with each other, as they are connected by the soil. Nowadays numerical simulation of soil structure interaction is of great interest and is applied to very different problems. These include for example the construction of reliable earthquake-resistant structures in seismic active areas, and also the increase of comfort of buildings by decouple them form surrounding emissions like vibrations induced by traffic of machine foundations.

This work shows that the simulation of soil-structure-interaction taking unbounded domains into account, which fulfils the Sommerfeld radiation condition exactly, is not only possible for academic examples, but for large scale real life problems as well. Therefore two numerical methods where coupled to create an efficient coupled method, which can be used to simulate soil-structure-interaction in time domain. The numerical implementation of this coupled approach bases on a combination of finite element method [1] and scaled boundary finite element method [2]. The finite element method is used to discretise the near-field, containing structures and its surrounding soil. The coupled infinite half-space, the far-field is realised by the scaled boundary finite element method.

A contemporary parallel implementation of the coupling algorithms is done, since the simulation of soil structure interaction in time domain is very time and memory consuming [3]. Subsequent the numerical performance of the implemented software is discussed

in terms of speed-up and efficiency. Different geotechnical applications are illustrated and the applicability of the coupled method is shown and discussed on chosen examples.

1 INTRODUCTION

Accurate simulations of wave-propagation is essential for soil-structure interaction (SSI) analysis. Unbounded domains or infinite half-spaces, require careful analysis and call for efficient methods in order to model wave-propagation to infinity. Whenever vibrations or impulses are emitted to soil, they induce waves traveling through the ground that can provoke structures to vibrate and even to fail. It is essential not only to analyze the structure itself but also to take the surrounding soil into account [4, 5, 6].

Different methods considering the surrounding unbounded domain by a transmitting boundary, have been developed during the last years. One of the simplest transmitting boundary conditions is a viscous boundary condition, acting like a dashpot [7]. Other local, arbitrary order absorbing boundary conditions [8, 8, 9] and several other types of transmitting boundaries (e.g., infinite elements [10, 11]) have been proposed, but none of them is able to fulfill the radiation condition exactly.

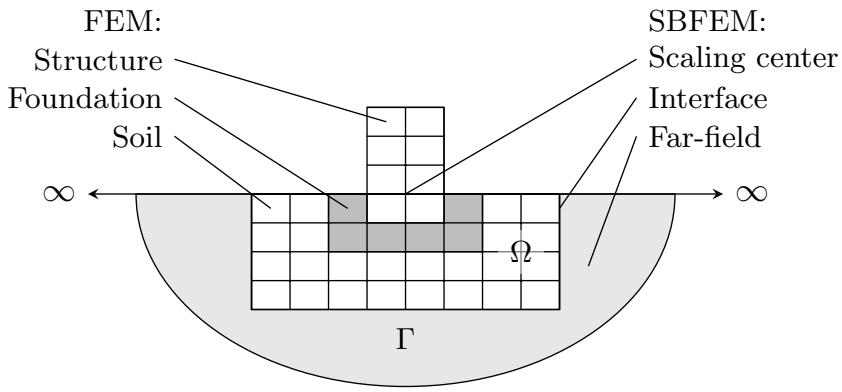

Figure 1: Problem definition.

Here a substructuring method is used (see Fig. 1) therefore the soil-structure system is spit up into a near-field and a far-field. Structure and foundation as well as parts of the surrounding soil are represent by the near-field Ω. The unbounded domain is represented by the far-field. The near-field can be easily discretized by finite elements the far-field is discretized by scaled boundary finite elements. Both methods are coupled at their common interface Γ. This discretisation assures that the radiation condition to infinity is fulfilled [2].

2 COUPLED FEM-SBFEM APPROACH

The equation of motion for the displacement-based FEM is given by

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{C}\dot{\mathbf{u}} + \mathbf{K}\mathbf{u} = \mathbf{p}, \tag{1}$$

where $\mathbf{M}$ is the mass matrix, $\mathbf{C}$ is the damping matrix and $\mathbf{K}$ the stiffness matrix. The vectors $\mathbf{u}$, its first and second derivative represents displacement, velocity and acceleration. Supposing that period T can be divided into n time steps of constant size yields to a time step length $\Delta t = \frac{T}{n}$. Applying the Newmark scheme [12] to equation (1) leads to

$$\mathbf{M}\ddot{\mathbf{u}}_{n+1} + \mathbf{C}\dot{\mathbf{u}}_{n+1} + \mathbf{K}\mathbf{u}_n = \mathbf{p}_{n+1}\,, \tag{2}$$

with the update rules for displacement

$$\mathbf{u}_{n+1} = \mathbf{u}_n + \Delta t\dot{\mathbf{u}}_n + \left(\frac{1}{2} - \beta\right)\Delta t^2\ddot{\mathbf{u}}_n + \beta\Delta t^2\ddot{\mathbf{u}}_{n+1}\,, \tag{3}$$

and velocity

$$\dot{\mathbf{u}}_{n+1} = \dot{\mathbf{u}}_n + (1-\gamma)\,\Delta t\ddot{\mathbf{u}}_n + \gamma\Delta t\ddot{\mathbf{u}}_{n+1}\,. \tag{4}$$

The parameters β and γ of the time step integration scheme should be set as follows

$$\gamma \geq \frac{1}{2} \quad \text{and} \quad \beta \geq \frac{1}{4}\left(\gamma + \frac{1}{2}\right)^2\,. \tag{5}$$

In order to couple FEM and SBFEM the entries of the matrices in equation (1) have to be reordered

$$\begin{bmatrix} \mathbf{M}_{\Omega\Omega} & \mathbf{M}_{\Omega\Gamma} \\ \mathbf{M}_{\Gamma\Omega} & \mathbf{M}_{\Gamma\Gamma} \end{bmatrix}\ddot{\mathbf{u}} + \begin{bmatrix} \mathbf{C}_{\Omega\Omega} & \mathbf{C}_{\Omega\Gamma} \\ \mathbf{C}_{\Gamma\Omega} & \mathbf{C}_{\Gamma\Gamma} \end{bmatrix}\dot{\mathbf{u}} + \begin{bmatrix} \mathbf{K}_{\Omega\Omega} & \mathbf{K}_{\Omega\Gamma} \\ \mathbf{K}_{\Gamma\Omega} & \mathbf{K}_{\Gamma\Gamma} \end{bmatrix}\mathbf{u} = \begin{bmatrix} \mathbf{p}_{\Omega\Omega} \\ \mathbf{p}_{\Gamma\Gamma} \end{bmatrix} - \begin{bmatrix} \mathbf{0} \\ \mathbf{p}_b \end{bmatrix} \tag{6}$$

so that the block with the subscript "ΩΩ" contains all nodes located in the near-field while the block with subscript "ΓΓ" contains all nodes at the far-field-interface. The blocks with the subscripts "ΩΓ" and "ΓΩ" include the coupling information of near-field and far-field nodes. The vector $\mathbf{p}_b$ acts on the boundare Γ only. This additional force describes the response of the infinite half-space and can be applied to the near-field as a load.

The near-field is described by the FEM the far-field by the SBFEM. The forces acting at the interface of near-field and far-field are given by the convolution integral

$$\mathbf{p}_b(t) = \int_0^t \mathbf{M}^\infty(t-\tau)\ddot{\mathbf{u}}(\tau)d\tau\,, \tag{7}$$

where $\mathbf{M}^{\infty}(t)$ is the unit-impulse matrix. To solve the convolution integral (7) the unit-impulse matrices $\mathbf{M}_i^{\infty}$ are assumed to be constant within the time step Δt,

$$\mathbf{M}^{\infty}(t) = \begin{cases} \mathbf{M}_0^{\infty} & t \in [0;\Delta t]\,, \\ \mathbf{M}_1^{\infty} & t \in [\Delta t;2\Delta t]\,, \\ \vdots & \vdots \\ \mathbf{M}_n^{\infty} & t \in [(n-1)\Delta t;n\Delta t]\ . \end{cases} \tag{8}$$

Due to this assumption and applying the time step integration scheme, equation (7) can be rewritten as

$$\mathbf{p}_b(t_n) = \gamma\Delta t\mathbf{M}_0^{\infty}\ddot{\mathbf{u}}_n + \sum_{j=1}^{n-1}\mathbf{M}_{n-j}^{\infty}\left(\dot{\mathbf{u}}_j - \dot{\mathbf{u}}_{j-1}\right). \tag{9}$$

The coupling of FEM and SBFEM is realized by simply adding equation (9) to the presorted FEM (6)

$$\begin{aligned} &\begin{bmatrix} \mathbf{M}_{\Omega\Omega} & \mathbf{M}_{\Omega\Gamma} \\ \mathbf{M}_{\Gamma\Omega} & \mathbf{M}_{\Gamma\Gamma} + \gamma\Delta t\mathbf{M}_0^{\infty} \end{bmatrix}\ddot{\mathbf{u}} + \begin{bmatrix} \mathbf{C}_{\Omega\Omega} & \mathbf{C}_{\Omega\Gamma} \\ \mathbf{C}_{\Gamma\Omega} & \mathbf{C}_{\Gamma\Gamma} \end{bmatrix}\dot{\mathbf{u}} + \\ &\begin{bmatrix} \mathbf{K}_{\Omega\Omega} & \mathbf{K}_{\Omega\Gamma} \\ \mathbf{K}_{\Gamma\Omega} & \mathbf{K}_{\Gamma\Gamma} \end{bmatrix}\mathbf{u} = \begin{bmatrix} \mathbf{p}_{\Omega\Omega} \\ \mathbf{p}_{\Gamma\Gamma} - \sum_{j=1}^{n-1}\mathbf{M}_{n-j}^{\infty}\left(\dot{\mathbf{u}}_j - \dot{\mathbf{u}}_{j-1}\right) \end{bmatrix}, \end{aligned} \tag{10}$$

so that the coupling is described in equation (10) completely.

3 IMPLEMENTATION

For the SSI analysis two programs are developed. The far-field is computed by ScaBo the coupled FEM-SBFEM analysis is computed by elPaSo [13, 14]. A more detailed description about the concept of ScaBo is published in [3]. Different third party libraries are used to achieve reasonable performance. ScaBo has to handle dense and sparse matrices as well, that is why ScaLAPACK builds the main core of this program [15]. Additional libraries like LAPACK, BLAS, BLACS, SLICOT, PLICOC (see Fig. 2(left)) are used to gain an optimal performance [16, 17, 18, 19].

The coupling of near-field and far-field is performed using elPaSo, here PETSc [20] builts the main core, hence sparse matrices are used. The libraries linkt to elPaSo are shown in Fig. 2 (right). To get access to direct solvers for linear equation systems MUMPS and SuperLU dist are also liked to elPaSo [21, 22, 23].

The communication is done by using the message passing interface (MPI) [24]. This allows to run both programs on distributed memory systems like clusters of computers and shared memory systems like multi core compute servers as well. This implementations guarantee a high flexibility regarding the utilized compute resources.

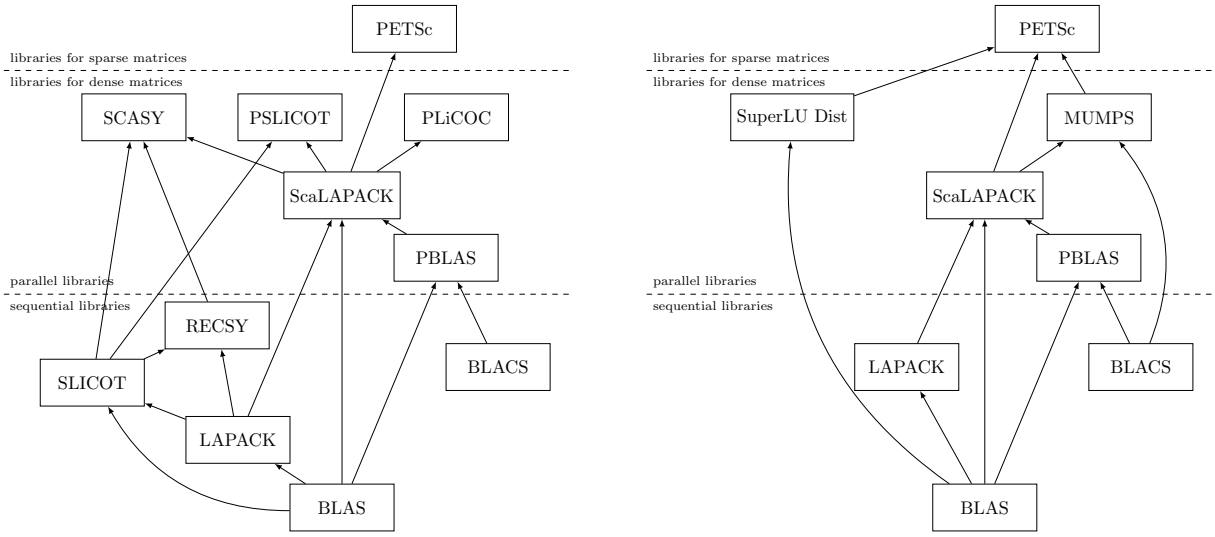

Figure 2: External libraries linked to ScaBo (left) elPaSo (right).

4 SETTLEMENT SIMULATION

In order to validate the implemented algorithms as illustrated in section 2 a settlement problem is carried out. Therefore two different discretisations of the near-field far-field interface are analysed. As already discussed in section 1, the problem has to be split into two separate domains. The near-field is represented by linear finite elements and the far-field by linear scaled boundary finite elements.

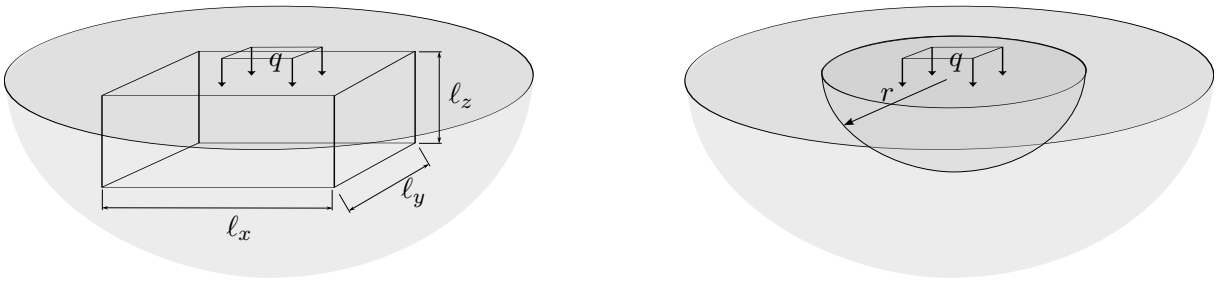

Figure 3: Infinite half space under constant area load q. (left) The computational domain is realized using a hemisphere. (right) The computational domain is realized using a cuboid.

Isotropic, homogeneous and fully linear elastic material is assumed for near-field and far-field as well. The material parameter are chosen as follows: Young's modulus $E = 21000$ [kNm^{-2}], Poisson ratio $\nu = 0.15$ and density $\rho = 2100$ [kgm^{-3}]. In this case a semi-analytical solution is available [25].

The numerical model is implemented for time domain analysis so that the displacements are time-dependent. As the coupled FEM-SBFEM approach fulfills the radiation condition, the nodal displacements become constant after a certain time, whenever con-

stant loads are applied. In those cases the numerical results approximate the static semi-analytical solution. To show the accuracy of the method we use two different meshing strategies, which are discussed in detail in the following sections.

One interface is discretised by a cubuid shaped mesh (CSM) see Fig. 3(left). The dimensions of the mesh are set to $\ell_x = \ell_y = 457.2$ [m] and $\ell_z = 190.5$ [m]. The second interface is discretised by a hemisphere shaped mesh (HSM) see Fig. 3(right). The distance between this centre and all interface nodes at the boundary Γ is exactly $r = 190.5$ [m]. The scaling centre is located in both cases in the middle of the loaded area, so that the smallest distance between scaling centre and boundary Γ is 190.5 [m]. On a square region of 152.4×152.4 [m^2] an area load $q = 70$ [kNm^{-2}] is applied. Table 1 summarizes the degrees of freedom (DoF) for the two discretised meshes.

Table 1: Discretisations with different number of DoF.

	$\mathrm{DoF}_{\mathrm{FEM}}$	$\mathrm{DoF}_{\mathrm{SBFEM}}$	$\Theta \frac{\mathrm{DoF}_{\mathrm{FEM}}}{\mathrm{DoF}_{\mathrm{SBFEM}}}$
CSM	14520	3720	25.6%
HSM	49155	3603	7.3%

The chosen material parameters yield to a longitudinal wave speed $c_p = 102.001$ ms^{-1}. So that the critical time step length is given by $\Delta t = \frac{r}{30 c_p} \approx 0.06$ s [26]. The parameters of Newmarks time step integration scheme are set to $\beta = 0.3025$ and $\gamma = 0.6$.

Evaluating the mentioned semi-analytical solution lead to a constant displacement $d(z = 0) = 0.560443138$ [m] in the centre of the loaded area directly at the near-fields surface. This solution is used to normalize the numerical time dependent solution as shown in Fig. 4.

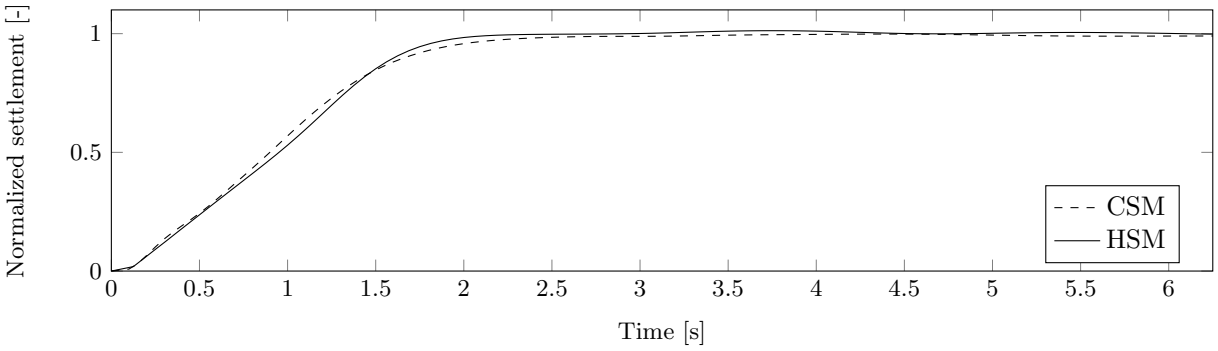

Figure 4: Time dependent normalized settlement.

5 FAR-FIELD MODEL REDUCTION

Computation of the unit-impulse response matrices $\mathbf{M}^{\infty}$ and the convolution integral eq. (7) is very expensive and memory consuming. Hence different model reductions are used to reduce the computational effort and memory consumption.

Since $\mathbf{M}^{\infty}$ increases after a certain time step t_m ($t_0 \leq t_m \leq t_n$), with a constant increment, not all n matrices have to be provided for the numerical simulation. For a simulation with n time steps, it is necessary to compute at least m matrices. All other matrices can be computed by a recursive algorithm [27].

Applying this model reduction to the meshes discussed in section 4 lead to the results as shown is Fig. 5. It is obvious that the used memory as well as the needed computation time decreases with the reduction of provided $\mathbf{M}^{\infty}$ matrices. As long as the number of $\mathbf{M}^{\infty}$ matrices is big enough the simulations relative error stays constant. If the number of precomputed matrices is too small the relative error increases. With $t_m = 50$ the relative error and also the solution is the same, compared to the reference solution without matrix extrapolation, but the memory usage is reduced to 32% and the run time is reduced to 37%.

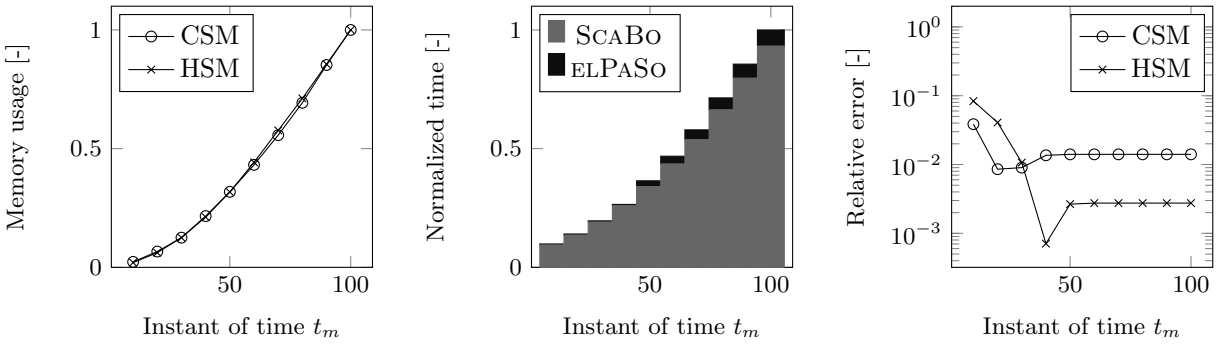

Figure 5: Memory usage (left), normalized compute time (mid) and relative error (right) when matrices $\mathbf{M}_t^{\infty}$, $t \leq m$ are extrapolated.

To make use of matrix extrapolation for model reduction, a recursive algorithm has to be implemented. Another approach without modifying the calculus is the geometrical decoupling. Therefore a threshold ϵ_z is introduced. This value is used to compare the entities in the unit-impulse response matrices. If the matrix entry is smaller than ϵ_z the influence of the corresponding nodes is supposed to be very small and so the value is not used for further computation. The influence of ϵ_z is studied within a range $10^{-8} \leq \epsilon_z \leq 10^{-2}$. The influence regarding the time used for computation is rather small, but the memory consumption is significant (see Fig.6). The fully populated matrices are computed and only at very end of the far-field computation the model is reduced. This reduced model is then used for later near-field far-field coupling, regarding the solutions relative error this model reduction is very robust.

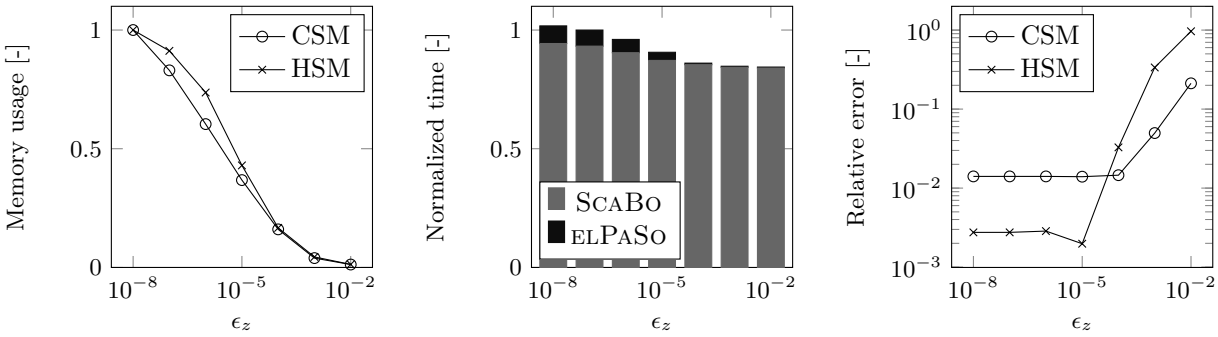

Figure 6: Memory usage (left), normalized compute time (mid) and relative error (right) when geometrical threshold ϵ_z is used.

The next step is consequently to decouple the far-field nodes not at the end of the far-field computation, but in the beginning. Hence the far-field is separated into an arbitrary number of substructures. Each substructure is solved separately. When the coupling is carried out, the substructures are combined with the near-field, so that the the complete far-field is considered. Memory usage, needed computational time and relative error are shown in Fig. 7. This reduction is very efficient in terms of memory and time consumption, but the relative error increases imminently.

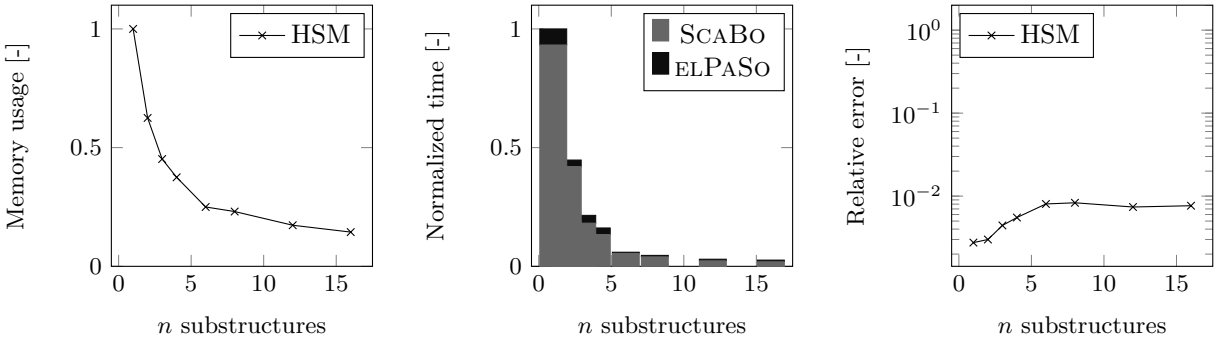

Figure 7: Memory usage (left), normalized compute time (mid) and relative error (right) when far-field is split up into substructures.

All three model reductions can be combined. The largest relative error of one method is dominating the other two. If substructuring is done carefully and the parameters t_m and ϵ_z are chosen in a valid range a major reduction of memory usage and computational time can be achieved. Here for instance if HSM is analysed with $t_m = 50$, $\epsilon_z = 10^{-5}$ and the far-field is divided into four substructures only 11% of memory is required and only 16.2% of the computational time. The solution is the same as the one which uses 4 substructures only.

6 PARALLEL PERFORMANCE

Desktop computer and even compute server are limited in memory and performance, hence the usage of compute cluster is a way to provide lots of memory and handle large numbers of unknowns in order to simulate realistic problems. As already mentioned both presented programs are designed to perform on distributed memory systems. In order to show the parallel performance a weak scaling is done. According to Amdahl [28] the time T needed to solve a certain problem can be split into one part which is going to be performed sequentially αT and another part which can be done in parallel $(1-\alpha)\,T$:

$$T = \alpha T + (1-\alpha)\,T\,. \tag{11}$$

Applying this to p processes lead to

$$T(p) = \alpha T(1) + \frac{(1-\alpha)\,T(1)}{p}\,. \tag{12}$$

Here the parallel performance of the coupled simulation is analysed. A detailed study of the parallel performance of ScaBo can be found in [3]. The weak scaling is conducted on a compute cluster of 15 nodes, each of them with 2 AMD Opteron 240 processors at 1.4 GHz and 3 GB of RAM, linked with a Myrinet-2000 interconnect. In all runs, one MPI rank per CPU is used (for instance, runs with 2 processes use only one node).

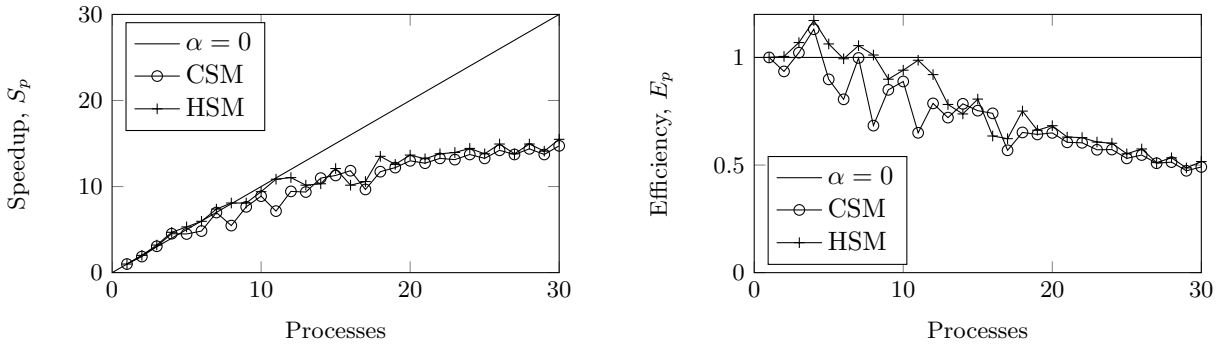

Figure 8: Parallel performence of the coupled simulation. Speedup (left), efficiency (right).

Fig. 8 displays the parallel speedup, which is defined as follows:

$$S_p = \frac{T(1)}{T(p)} \tag{13}$$

The parallel speedup is equal to the gain factor when the code runs with p processors. The plot shows an increasing of the speedup, which does not decay until 30 processes are employed. The speedup is not very close to the theoretical maximum $\alpha = 0$, but it shows a significant gain of about 15 with 30 processes. In terms of efficiency

$$E_p = \frac{S_p}{p} = \frac{T(1)}{p \cdot T(p)}\,, \tag{14}$$

which gives an indication of how well the code utilizes the available processors, a level of 50% is achieved in the worst case.

7 CONCLUSIONS

It is shown, that different model reductions techniques applied to the far-field can reduce the required memory and computational time without a leak of accuracy. Both, model reduction techniques and parallel computing can provide an answer to the high computational cost resulting from the application of SBFEM to very large scaled engineering problems. It is demonstrated that, in practice, when run on a moderately-sized cluster, the coupled FEM-SBFEM approach can yield the solution not only to academic benchmarks, but also to real problems. More complex, large-scale applications using this coupling approach will follow.

REFERENCES

[1] Zienkiewicz, O.C and Taylor, R.C, *The finite element method*. 4th Edition, Vol. 1, McGraw Hill, (1989).

[2] Wolf, J. and Song, C. *Finite-Element Modelling of Unbounded Media*, John Wiley & Sons, Chichester, (1996).

[3] Schauer, M; Roman, J.E.; Quintana-Ortí, E.S. and Langer, S. Parallel Computation of 3-D Soil-Structure Interaction in Time Domain with a Coupled FEM/SBFEM Approach. *Journal of Scientific Computing*, (2012), **52**:446–467.

[4] Antes, H. and Spyrakos, C. Soil-structure interaction. In: Beskos D and Anagnotopou los S. (eds) *Computer Analysis and Design of Earthquake Resistant Structures*, Computational Mechanics Publications, Southampton, (1997).

[5] Meskouris, K.; Hinzen, K.G.; Butenweg, C. and Mistler, M. *Bauwerke und Erdbeben - Grundlagen - Anwendung - Beispiele*. Vieweg+Teubner Verlag, Wiesbaden, (2007).

[6] Petersen C. *Dynamik der Baukonstruktionen*. Vieweg & Sohn Verlagsge- sellschaft mbH, Braunschweig/Wiesbaden, (2000).

[7] Lysmer, J. and Kuhlmeyer, R.L. Finite dynamic model for infinite media. *Journal of Engineering Mechanics*, **95**:859–875, (1969).

[8] Engquist, B. and Majda, A. Absorbing boundary conditions for the numerical simulation of waves. *Mathematics of Computation*, **31**:629–651, (1977).

[9] Liao, Z.P. and Wong, H.L. A transmitting boundary for the numerical simulation of elastic wave propagation. *Soil Dynamics and Earthquake Engineering*, **3**:174183, (1984).

[10] Bettess, P. *Infinite Elements*. Penshaw Press, Sunderland, U.K. (1992).

[11] Astley, R.J. Infinite elements for wave problems: a review of current formulations and a assessment of accuracy. *International Journal for Numerical Methods in Engineering*, **49**:951976, (2000).

[12] Newmark, N. A method of computation for structural dynamics. *Journal of Engineering Mechanics Division*, (1959), **85**:67–94.

[13] Schauer, M. and Roman, J.E. ScaBo Manual, Revision: 369, TU Braunschweig, Institut für Angewandte Mechanik (2010–2013), Institut für Konstruktionstechnik (2013), (2010–2014).

[14] Beck, S.; Clasen, D.; Lehmann, L.; Rurkowska, K.; Schauer, M. and Wulkau, M. elPaSo Manual, Revision: 433, TU Braunschweig, Institut für Angewandte Mechanik (2008–2013), Institut für Konstruktionstechnik (2013), (2008–2014).

[15] Blackford, L.S.; Choi, J.; Cleary, A.; DAzevedo, E.; Demmel, J.; Dhillon, I.; Dongarra, J.; Hammarling, S.; Henry, G.; Petitet, A.; Stanley, K.; Walker, D. and Whaley, R.C. *ScaLAPACK Users Guide*. Society for Industrial and Applied Mathematics, Philadelphia, (1997).

[16] Anderson, E.; Bai, Z.; Bischof, C.; Demmel, J.; Dongarra, J.; Croz, J.D.; Greenbaum, A.; Hammarling, S.; McKenney, A. and Sorensen, D. *LAPACK Users Guide*. Society for Industrial and Applied Mathematics, Philadelphia, (1992).

[17] Dongarra, J.J. and Whaley, R.C. *LAPACK working note 94: A users guide to the BLACS v1.1. Tech. Rep. UT-CS-95-281*, Department of Computer Science, University of Tennessee, (1995).

[18] Benner, P.; Quintana-Ortí, E.S. and Quintana-Ortí, G. Solving linear-quadratic optimal control problems on parallel computers. *Optimization Methods and Software*, **23**:879909, (2008).

[19] Guerrero, D.; Hernández, V. and Román, J.E. *Parallel SLICOT model reduction routines: The Cholesky factor of Grammians*. In: Proceedings of the 15th Triennal IFAC World Congress, Barcelona, Spain, (2002).

[20] Balay, S.; Buschelman, K.; Eijkhout, V.; Gropp, W.D.; Kaushik, D.; Knepley, M.; McInnes, L.C.; Smith, B.F. and Zhang, H. *PETSc users manual. Tech. Rep. ANL-95/11 - Revision 3.1*, Argonne National Laboratory, (2010).

[21] MUMPS Team. *MUMPS - a MUltifrontal Massively Parallel sparse direct Solver*, http://graal.ens-lyon.fr/MUMPS, (2012).

[22] Li, X.S. An Overview of SuperLU: Algorithms, Implementation, and User Interface. *Transactions on Mathematical Software*, **31**:302–325, (2005).

[23] Li, X.S.; Demmel, J.W.; Gilbert, J.R.; Grigori, I.L.; Shao, M. and Yamazaki, I. *SuperLU Users' Guide*, LBNL-44289, Lawrence Berkeley National Laboratory, (1999).

[24] MPI Forum (1994) *The message passing interface (MPI) standard*, http://www.mcs.anl.gov/mpi

[25] Harr, M.E. *Foundations of Theoretical Soil Mechanics*. McGraw-Hill Book Company, New York, (1966).

[26] Borsutzky, R. *Seismic Risk Analysis of Buried Lifelines*. Technische Universität Braunschweig, Braunschweiger Schriften zur Mechanik, 63, (2008).

[27] Lehmann, L. An effective finite element approach for soil-structure analysis in the time-domain, *Structural Engineering and Mechanics*, (2005), **21**:437–450.

[28] Amdahl, G.M. Validity of the single processor approach to achieving large-scale computing capabilities. *AFIPS Conference Proceedings*, (1967), **30**:483–485.